| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/DefaultListModel.html) | [**Tree**](http://docs.google.com/package-tree.html) | [**Deprecated**](http://docs.google.com/deprecated-list.html) | [**Index**](http://docs.google.com/index-files/index-1.html) | [**Help**](http://docs.google.com/help-doc.html) | | --- | --- | --- | --- | --- | --- | --- | --- | | | ***Java™ Platform***  ***Standard Ed. 6*** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [**PREV CLASS**](http://docs.google.com/javax/swing/DefaultListCellRenderer.UIResource.html)   [**NEXT CLASS**](http://docs.google.com/javax/swing/DefaultListSelectionModel.html) | [**FRAMES**](http://docs.google.com/index.html?javax/swing/DefaultListModel.html)    [**NO FRAMES**](http://docs.google.com/DefaultListModel.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | [FIELD](#2et92p0) | [CONSTR](#tyjcwt) | [METHOD](#3dy6vkm) | DETAIL: FIELD | [CONSTR](#2s8eyo1) | [METHOD](#3rdcrjn) |

## **javax.swing**

Class DefaultListModel

[java.lang.Object](http://docs.google.com/java/lang/Object.html)  
 [javax.swing.AbstractListModel](http://docs.google.com/javax/swing/AbstractListModel.html)  
 **javax.swing.DefaultListModel**

**All Implemented Interfaces:** [Serializable](http://docs.google.com/java/io/Serializable.html), [ListModel](http://docs.google.com/javax/swing/ListModel.html)

public class **DefaultListModel**extends [AbstractListModel](http://docs.google.com/javax/swing/AbstractListModel.html)

This class loosely implements the java.util.Vector API, in that it implements the 1.1.x version of java.util.Vector, has no collection class support, and notifies the ListDataListeners when changes occur. Presently it delegates to a Vector, in a future release it will be a real Collection implementation.

**Warning:** Serialized objects of this class will not be compatible with future Swing releases. The current serialization support is appropriate for short term storage or RMI between applications running the same version of Swing. As of 1.4, support for long term storage of all JavaBeansTM has been added to the java.beans package. Please see [XMLEncoder](http://docs.google.com/java/beans/XMLEncoder.html).

| **Field Summary** | |
| --- | --- |

| **Fields inherited from class javax.swing.**[**AbstractListModel**](http://docs.google.com/javax/swing/AbstractListModel.html) |
| --- |
| [listenerList](http://docs.google.com/javax/swing/AbstractListModel.html#listenerList) |

| **Constructor Summary** | |
| --- | --- |
| [**DefaultListModel**](http://docs.google.com/javax/swing/DefaultListModel.html#DefaultListModel())() |

| **Method Summary** | |
| --- | --- |
| void | [**add**](http://docs.google.com/javax/swing/DefaultListModel.html#add(int,%20java.lang.Object))(int index, [Object](http://docs.google.com/java/lang/Object.html) element)            Inserts the specified element at the specified position in this list. |
| void | [**addElement**](http://docs.google.com/javax/swing/DefaultListModel.html#addElement(java.lang.Object))([Object](http://docs.google.com/java/lang/Object.html) obj)            Adds the specified component to the end of this list. |
| int | [**capacity**](http://docs.google.com/javax/swing/DefaultListModel.html#capacity())()            Returns the current capacity of this list. |
| void | [**clear**](http://docs.google.com/javax/swing/DefaultListModel.html#clear())()            Removes all of the elements from this list. |
| boolean | [**contains**](http://docs.google.com/javax/swing/DefaultListModel.html#contains(java.lang.Object))([Object](http://docs.google.com/java/lang/Object.html) elem)            Tests whether the specified object is a component in this list. |
| void | [**copyInto**](http://docs.google.com/javax/swing/DefaultListModel.html#copyInto(java.lang.Object%5B%5D))([Object](http://docs.google.com/java/lang/Object.html)[] anArray)            Copies the components of this list into the specified array. |
| [Object](http://docs.google.com/java/lang/Object.html) | [**elementAt**](http://docs.google.com/javax/swing/DefaultListModel.html#elementAt(int))(int index)            Returns the component at the specified index. |
| [Enumeration](http://docs.google.com/java/util/Enumeration.html)<?> | [**elements**](http://docs.google.com/javax/swing/DefaultListModel.html#elements())()            Returns an enumeration of the components of this list. |
| void | [**ensureCapacity**](http://docs.google.com/javax/swing/DefaultListModel.html#ensureCapacity(int))(int minCapacity)            Increases the capacity of this list, if necessary, to ensure that it can hold at least the number of components specified by the minimum capacity argument. |
| [Object](http://docs.google.com/java/lang/Object.html) | [**firstElement**](http://docs.google.com/javax/swing/DefaultListModel.html#firstElement())()            Returns the first component of this list. |
| [Object](http://docs.google.com/java/lang/Object.html) | [**get**](http://docs.google.com/javax/swing/DefaultListModel.html#get(int))(int index)            Returns the element at the specified position in this list. |
| [Object](http://docs.google.com/java/lang/Object.html) | [**getElementAt**](http://docs.google.com/javax/swing/DefaultListModel.html#getElementAt(int))(int index)            Returns the component at the specified index. |
| int | [**getSize**](http://docs.google.com/javax/swing/DefaultListModel.html#getSize())()            Returns the number of components in this list. |
| int | [**indexOf**](http://docs.google.com/javax/swing/DefaultListModel.html#indexOf(java.lang.Object))([Object](http://docs.google.com/java/lang/Object.html) elem)            Searches for the first occurrence of elem. |
| int | [**indexOf**](http://docs.google.com/javax/swing/DefaultListModel.html#indexOf(java.lang.Object,%20int))([Object](http://docs.google.com/java/lang/Object.html) elem, int index)            Searches for the first occurrence of elem, beginning the search at index. |
| void | [**insertElementAt**](http://docs.google.com/javax/swing/DefaultListModel.html#insertElementAt(java.lang.Object,%20int))([Object](http://docs.google.com/java/lang/Object.html) obj, int index)            Inserts the specified object as a component in this list at the specified index. |
| boolean | [**isEmpty**](http://docs.google.com/javax/swing/DefaultListModel.html#isEmpty())()            Tests whether this list has any components. |
| [Object](http://docs.google.com/java/lang/Object.html) | [**lastElement**](http://docs.google.com/javax/swing/DefaultListModel.html#lastElement())()            Returns the last component of the list. |
| int | [**lastIndexOf**](http://docs.google.com/javax/swing/DefaultListModel.html#lastIndexOf(java.lang.Object))([Object](http://docs.google.com/java/lang/Object.html) elem)            Returns the index of the last occurrence of elem. |
| int | [**lastIndexOf**](http://docs.google.com/javax/swing/DefaultListModel.html#lastIndexOf(java.lang.Object,%20int))([Object](http://docs.google.com/java/lang/Object.html) elem, int index)            Searches backwards for elem, starting from the specified index, and returns an index to it. |
| [Object](http://docs.google.com/java/lang/Object.html) | [**remove**](http://docs.google.com/javax/swing/DefaultListModel.html#remove(int))(int index)            Removes the element at the specified position in this list. |
| void | [**removeAllElements**](http://docs.google.com/javax/swing/DefaultListModel.html#removeAllElements())()            Removes all components from this list and sets its size to zero. |
| boolean | [**removeElement**](http://docs.google.com/javax/swing/DefaultListModel.html#removeElement(java.lang.Object))([Object](http://docs.google.com/java/lang/Object.html) obj)            Removes the first (lowest-indexed) occurrence of the argument from this list. |
| void | [**removeElementAt**](http://docs.google.com/javax/swing/DefaultListModel.html#removeElementAt(int))(int index)            Deletes the component at the specified index. |
| void | [**removeRange**](http://docs.google.com/javax/swing/DefaultListModel.html#removeRange(int,%20int))(int fromIndex, int toIndex)            Deletes the components at the specified range of indexes. |
| [Object](http://docs.google.com/java/lang/Object.html) | [**set**](http://docs.google.com/javax/swing/DefaultListModel.html#set(int,%20java.lang.Object))(int index, [Object](http://docs.google.com/java/lang/Object.html) element)            Replaces the element at the specified position in this list with the specified element. |
| void | [**setElementAt**](http://docs.google.com/javax/swing/DefaultListModel.html#setElementAt(java.lang.Object,%20int))([Object](http://docs.google.com/java/lang/Object.html) obj, int index)            Sets the component at the specified index of this list to be the specified object. |
| void | [**setSize**](http://docs.google.com/javax/swing/DefaultListModel.html#setSize(int))(int newSize)            Sets the size of this list. |
| int | [**size**](http://docs.google.com/javax/swing/DefaultListModel.html#size())()            Returns the number of components in this list. |
| [Object](http://docs.google.com/java/lang/Object.html)[] | [**toArray**](http://docs.google.com/javax/swing/DefaultListModel.html#toArray())()            Returns an array containing all of the elements in this list in the correct order. |
| [String](http://docs.google.com/java/lang/String.html) | [**toString**](http://docs.google.com/javax/swing/DefaultListModel.html#toString())()            Returns a string that displays and identifies this object's properties. |
| void | [**trimToSize**](http://docs.google.com/javax/swing/DefaultListModel.html#trimToSize())()            Trims the capacity of this list to be the list's current size. |

| **Methods inherited from class javax.swing.**[**AbstractListModel**](http://docs.google.com/javax/swing/AbstractListModel.html) |
| --- |
| [addListDataListener](http://docs.google.com/javax/swing/AbstractListModel.html#addListDataListener(javax.swing.event.ListDataListener)), [fireContentsChanged](http://docs.google.com/javax/swing/AbstractListModel.html#fireContentsChanged(java.lang.Object,%20int,%20int)), [fireIntervalAdded](http://docs.google.com/javax/swing/AbstractListModel.html#fireIntervalAdded(java.lang.Object,%20int,%20int)), [fireIntervalRemoved](http://docs.google.com/javax/swing/AbstractListModel.html#fireIntervalRemoved(java.lang.Object,%20int,%20int)), [getListDataListeners](http://docs.google.com/javax/swing/AbstractListModel.html#getListDataListeners()), [getListeners](http://docs.google.com/javax/swing/AbstractListModel.html#getListeners(java.lang.Class)), [removeListDataListener](http://docs.google.com/javax/swing/AbstractListModel.html#removeListDataListener(javax.swing.event.ListDataListener)) |

| **Methods inherited from class java.lang.**[**Object**](http://docs.google.com/java/lang/Object.html) |
| --- |
| [clone](http://docs.google.com/java/lang/Object.html#clone()), [equals](http://docs.google.com/java/lang/Object.html#equals(java.lang.Object)), [finalize](http://docs.google.com/java/lang/Object.html#finalize()), [getClass](http://docs.google.com/java/lang/Object.html#getClass()), [hashCode](http://docs.google.com/java/lang/Object.html#hashCode()), [notify](http://docs.google.com/java/lang/Object.html#notify()), [notifyAll](http://docs.google.com/java/lang/Object.html#notifyAll()), [wait](http://docs.google.com/java/lang/Object.html#wait()), [wait](http://docs.google.com/java/lang/Object.html#wait(long)), [wait](http://docs.google.com/java/lang/Object.html#wait(long,%20int)) |

| **Constructor Detail** |
| --- |

### DefaultListModel

public **DefaultListModel**()

| **Method Detail** |
| --- |

### getSize

public int **getSize**()

Returns the number of components in this list.

This method is identical to size, which implements the List interface defined in the 1.2 Collections framework. This method exists in conjunction with setSize so that size is identifiable as a JavaBean property.

**Returns:**the number of components in this list**See Also:**[size()](http://docs.google.com/javax/swing/DefaultListModel.html#size())

### getElementAt

public [Object](http://docs.google.com/java/lang/Object.html) **getElementAt**(int index)

Returns the component at the specified index.**Note:** Although this method is not deprecated, the preferred method to use is get(int), which implements the List interface defined in the 1.2 Collections framework.

**Parameters:**index - an index into this list **Returns:**the component at the specified index **Throws:** [ArrayIndexOutOfBoundsException](http://docs.google.com/java/lang/ArrayIndexOutOfBoundsException.html) - if the index is negative or greater than the current size of this list**See Also:**[get(int)](http://docs.google.com/javax/swing/DefaultListModel.html#get(int))

### copyInto

public void **copyInto**([Object](http://docs.google.com/java/lang/Object.html)[] anArray)

Copies the components of this list into the specified array. The array must be big enough to hold all the objects in this list, else an IndexOutOfBoundsException is thrown.

**Parameters:**anArray - the array into which the components get copied**See Also:**[Vector.copyInto(Object[])](http://docs.google.com/java/util/Vector.html#copyInto(java.lang.Object%5B%5D))

### trimToSize

public void **trimToSize**()

Trims the capacity of this list to be the list's current size.

**See Also:**[Vector.trimToSize()](http://docs.google.com/java/util/Vector.html#trimToSize())

### ensureCapacity

public void **ensureCapacity**(int minCapacity)

Increases the capacity of this list, if necessary, to ensure that it can hold at least the number of components specified by the minimum capacity argument.

**Parameters:**minCapacity - the desired minimum capacity**See Also:**[Vector.ensureCapacity(int)](http://docs.google.com/java/util/Vector.html#ensureCapacity(int))

### setSize

public void **setSize**(int newSize)

Sets the size of this list.

**Parameters:**newSize - the new size of this list**See Also:**[Vector.setSize(int)](http://docs.google.com/java/util/Vector.html#setSize(int))

### capacity

public int **capacity**()

Returns the current capacity of this list.

**Returns:**the current capacity**See Also:**[Vector.capacity()](http://docs.google.com/java/util/Vector.html#capacity())

### size

public int **size**()

Returns the number of components in this list.

**Returns:**the number of components in this list**See Also:**[Vector.size()](http://docs.google.com/java/util/Vector.html#size())

### isEmpty

public boolean **isEmpty**()

Tests whether this list has any components.

**Returns:**true if and only if this list has no components, that is, its size is zero; false otherwise**See Also:**[Vector.isEmpty()](http://docs.google.com/java/util/Vector.html#isEmpty())

### elements

public [Enumeration](http://docs.google.com/java/util/Enumeration.html)<?> **elements**()

Returns an enumeration of the components of this list.

**Returns:**an enumeration of the components of this list**See Also:**[Vector.elements()](http://docs.google.com/java/util/Vector.html#elements())

### contains

public boolean **contains**([Object](http://docs.google.com/java/lang/Object.html) elem)

Tests whether the specified object is a component in this list.

**Parameters:**elem - an object **Returns:**true if the specified object is the same as a component in this list**See Also:**[Vector.contains(Object)](http://docs.google.com/java/util/Vector.html#contains(java.lang.Object))

### indexOf

public int **indexOf**([Object](http://docs.google.com/java/lang/Object.html) elem)

Searches for the first occurrence of elem.

**Parameters:**elem - an object **Returns:**the index of the first occurrence of the argument in this list; returns -1 if the object is not found**See Also:**[Vector.indexOf(Object)](http://docs.google.com/java/util/Vector.html#indexOf(java.lang.Object))

### indexOf

public int **indexOf**([Object](http://docs.google.com/java/lang/Object.html) elem,  
 int index)

Searches for the first occurrence of elem, beginning the search at index.

**Parameters:**elem - an desired componentindex - the index from which to begin searching **Returns:**the index where the first occurrence of elem is found after index; returns -1 if the elem is not found in the list**See Also:**[Vector.indexOf(Object,int)](http://docs.google.com/java/util/Vector.html#indexOf(java.lang.Object,%20int))

### lastIndexOf

public int **lastIndexOf**([Object](http://docs.google.com/java/lang/Object.html) elem)

Returns the index of the last occurrence of elem.

**Parameters:**elem - the desired component **Returns:**the index of the last occurrence of elem in the list; returns -1 if the object is not found**See Also:**[Vector.lastIndexOf(Object)](http://docs.google.com/java/util/Vector.html#lastIndexOf(java.lang.Object))

### lastIndexOf

public int **lastIndexOf**([Object](http://docs.google.com/java/lang/Object.html) elem,  
 int index)

Searches backwards for elem, starting from the specified index, and returns an index to it.

**Parameters:**elem - the desired componentindex - the index to start searching from **Returns:**the index of the last occurrence of the elem in this list at position less than index; returns -1 if the object is not found**See Also:**[Vector.lastIndexOf(Object,int)](http://docs.google.com/java/util/Vector.html#lastIndexOf(java.lang.Object,%20int))

### elementAt

public [Object](http://docs.google.com/java/lang/Object.html) **elementAt**(int index)

Returns the component at the specified index. Throws an ArrayIndexOutOfBoundsException if the index is negative or not less than the size of the list.**Note:** Although this method is not deprecated, the preferred method to use is get(int), which implements the List interface defined in the 1.2 Collections framework.

**Parameters:**index - an index into this list **Returns:**the component at the specified index**See Also:**[get(int)](http://docs.google.com/javax/swing/DefaultListModel.html#get(int)), [Vector.elementAt(int)](http://docs.google.com/java/util/Vector.html#elementAt(int))

### firstElement

public [Object](http://docs.google.com/java/lang/Object.html) **firstElement**()

Returns the first component of this list. Throws a NoSuchElementException if this vector has no components.

**Returns:**the first component of this list**See Also:**[Vector.firstElement()](http://docs.google.com/java/util/Vector.html#firstElement())

### lastElement

public [Object](http://docs.google.com/java/lang/Object.html) **lastElement**()

Returns the last component of the list. Throws a NoSuchElementException if this vector has no components.

**Returns:**the last component of the list**See Also:**[Vector.lastElement()](http://docs.google.com/java/util/Vector.html#lastElement())

### setElementAt

public void **setElementAt**([Object](http://docs.google.com/java/lang/Object.html) obj,  
 int index)

Sets the component at the specified index of this list to be the specified object. The previous component at that position is discarded.

Throws an ArrayIndexOutOfBoundsException if the index is invalid.

**Note:** Although this method is not deprecated, the preferred method to use is set(int,Object), which implements the List interface defined in the 1.2 Collections framework.

**Parameters:**obj - what the component is to be set toindex - the specified index**See Also:**[set(int,Object)](http://docs.google.com/javax/swing/DefaultListModel.html#set(int,%20java.lang.Object)), [Vector.setElementAt(Object,int)](http://docs.google.com/java/util/Vector.html#setElementAt(E,%20int))

### removeElementAt

public void **removeElementAt**(int index)

Deletes the component at the specified index.

Throws an ArrayIndexOutOfBoundsException if the index is invalid.

**Note:** Although this method is not deprecated, the preferred method to use is remove(int), which implements the List interface defined in the 1.2 Collections framework.

**Parameters:**index - the index of the object to remove**See Also:**[remove(int)](http://docs.google.com/javax/swing/DefaultListModel.html#remove(int)), [Vector.removeElementAt(int)](http://docs.google.com/java/util/Vector.html#removeElementAt(int))

### insertElementAt

public void **insertElementAt**([Object](http://docs.google.com/java/lang/Object.html) obj,  
 int index)

Inserts the specified object as a component in this list at the specified index.

Throws an ArrayIndexOutOfBoundsException if the index is invalid.

**Note:** Although this method is not deprecated, the preferred method to use is add(int,Object), which implements the List interface defined in the 1.2 Collections framework.

**Parameters:**obj - the component to insertindex - where to insert the new component **Throws:** [ArrayIndexOutOfBoundsException](http://docs.google.com/java/lang/ArrayIndexOutOfBoundsException.html) - if the index was invalid**See Also:**[add(int,Object)](http://docs.google.com/javax/swing/DefaultListModel.html#add(int,%20java.lang.Object)), [Vector.insertElementAt(Object,int)](http://docs.google.com/java/util/Vector.html#insertElementAt(E,%20int))

### addElement

public void **addElement**([Object](http://docs.google.com/java/lang/Object.html) obj)

Adds the specified component to the end of this list.

**Parameters:**obj - the component to be added**See Also:**[Vector.addElement(Object)](http://docs.google.com/java/util/Vector.html#addElement(E))

### removeElement

public boolean **removeElement**([Object](http://docs.google.com/java/lang/Object.html) obj)

Removes the first (lowest-indexed) occurrence of the argument from this list.

**Parameters:**obj - the component to be removed **Returns:**true if the argument was a component of this list; false otherwise**See Also:**[Vector.removeElement(Object)](http://docs.google.com/java/util/Vector.html#removeElement(java.lang.Object))

### removeAllElements

public void **removeAllElements**()

Removes all components from this list and sets its size to zero.**Note:** Although this method is not deprecated, the preferred method to use is clear, which implements the List interface defined in the 1.2 Collections framework.

**See Also:**[clear()](http://docs.google.com/javax/swing/DefaultListModel.html#clear()), [Vector.removeAllElements()](http://docs.google.com/java/util/Vector.html#removeAllElements())

### toString

public [String](http://docs.google.com/java/lang/String.html) **toString**()

Returns a string that displays and identifies this object's properties.

**Overrides:**[toString](http://docs.google.com/java/lang/Object.html#toString()) in class [Object](http://docs.google.com/java/lang/Object.html) **Returns:**a String representation of this object

### toArray

public [Object](http://docs.google.com/java/lang/Object.html)[] **toArray**()

Returns an array containing all of the elements in this list in the correct order.

**Returns:**an array containing the elements of the list**See Also:**[Vector.toArray()](http://docs.google.com/java/util/Vector.html#toArray())

### get

public [Object](http://docs.google.com/java/lang/Object.html) **get**(int index)

Returns the element at the specified position in this list.

Throws an ArrayIndexOutOfBoundsException if the index is out of range (index < 0 || index >= size()).

**Parameters:**index - index of element to return

### set

public [Object](http://docs.google.com/java/lang/Object.html) **set**(int index,  
 [Object](http://docs.google.com/java/lang/Object.html) element)

Replaces the element at the specified position in this list with the specified element.

Throws an ArrayIndexOutOfBoundsException if the index is out of range (index < 0 || index >= size()).

**Parameters:**index - index of element to replaceelement - element to be stored at the specified position **Returns:**the element previously at the specified position

### add

public void **add**(int index,  
 [Object](http://docs.google.com/java/lang/Object.html) element)

Inserts the specified element at the specified position in this list.

Throws an ArrayIndexOutOfBoundsException if the index is out of range (index < 0 || index > size()).

**Parameters:**index - index at which the specified element is to be insertedelement - element to be inserted

### remove

public [Object](http://docs.google.com/java/lang/Object.html) **remove**(int index)

Removes the element at the specified position in this list. Returns the element that was removed from the list.

Throws an ArrayIndexOutOfBoundsException if the index is out of range (index < 0 || index >= size()).

**Parameters:**index - the index of the element to removed

### clear

public void **clear**()

Removes all of the elements from this list. The list will be empty after this call returns (unless it throws an exception).

### removeRange

public void **removeRange**(int fromIndex,  
 int toIndex)

Deletes the components at the specified range of indexes. The removal is inclusive, so specifying a range of (1,5) removes the component at index 1 and the component at index 5, as well as all components in between.

Throws an ArrayIndexOutOfBoundsException if the index was invalid. Throws an IllegalArgumentException if fromIndex > toIndex.

**Parameters:**fromIndex - the index of the lower end of the rangetoIndex - the index of the upper end of the range**See Also:**[remove(int)](http://docs.google.com/javax/swing/DefaultListModel.html#remove(int))

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/DefaultListModel.html) | [**Tree**](http://docs.google.com/package-tree.html) | [**Deprecated**](http://docs.google.com/deprecated-list.html) | [**Index**](http://docs.google.com/index-files/index-1.html) | [**Help**](http://docs.google.com/help-doc.html) | | --- | --- | --- | --- | --- | --- | --- | --- | | | ***Java™ Platform***  ***Standard Ed. 6*** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [**PREV CLASS**](http://docs.google.com/javax/swing/DefaultListCellRenderer.UIResource.html)   [**NEXT CLASS**](http://docs.google.com/javax/swing/DefaultListSelectionModel.html) | [**FRAMES**](http://docs.google.com/index.html?javax/swing/DefaultListModel.html)    [**NO FRAMES**](http://docs.google.com/DefaultListModel.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | [FIELD](#2et92p0) | [CONSTR](#tyjcwt) | [METHOD](#3dy6vkm) | DETAIL: FIELD | [CONSTR](#2s8eyo1) | [METHOD](#3rdcrjn) |

[Submit a bug or feature](http://bugs.sun.com/services/bugreport/index.jsp)

For further API reference and developer documentation, see [Java SE Developer Documentation](http://docs.google.com/webnotes/devdocs-vs-specs.html). That documentation contains more detailed, developer-targeted descriptions, with conceptual overviews, definitions of terms, workarounds, and working code examples.

Copyright 2006 Sun Microsystems, Inc. All rights reserved. Use is subject to [license terms](http://docs.google.com/legal/license.html). Also see the [documentation redistribution policy](http://java.sun.com/docs/redist.html).